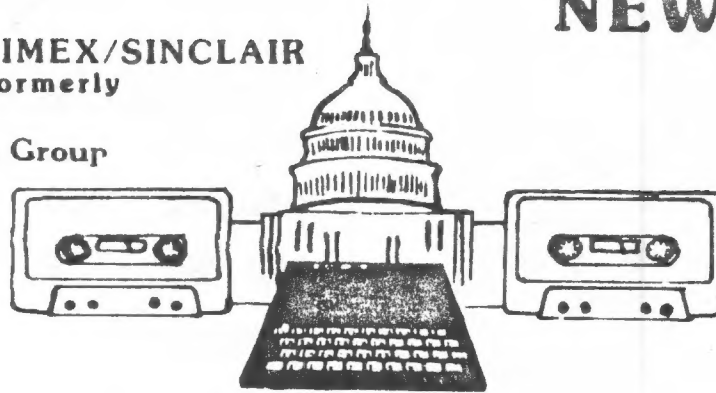


CATS

CAPITOL AREA TIMEX/SINCLAIR
USERS GROUP :Formerly
Prince George's
Timex/Sinclair User's Group

NEWSLETTER



July 1985
Vol. 3, No. 4

* * * CONTENTS * * *

→ Tape Dubbing	2	Poor Man's MODEM	11
Expanded Multiplication	3	Z80 Chip Organization	12
Machine Code Notes	3	MScript Tips	13
2K Calculator	4	Microdrive Review	15
Critical Path Method	5	Modulo!	16
Back-ups for VU-FILE	6	Binary Manipulation	17
Science Fair Timexers	7	New News	18
A Second SOURCE	10		

President's Column

Well, it's been a year that I've been serving as president of C.A.T.S. When I was elected to the post, I never dreamed that the User's Group would still be going strong one year hence. I thought there might be perhaps 10-15 diehards hanging around. Instead, we've got over 160 members, and a press run of 300 newsletters each month (in case you're wondering, the extras go out as complimentary copies to other User's Groups, manufacturers, and distribution at two local stores).

This strength has been due to the dedication of a number of longterm members, and to the acquisition of some active new members. Jules Gesang has been tireless in promoting the interests of the club, acting as contact for new members, and smoothing the flow of the newsletter through the Post Office. Perhaps his greatest coup was last month's newsletter delivery - 3 days!

There are many other members that deserve acknowledgement, but I'm afraid that singling out any more would slight the efforts of others. A look through the last few newsletters will show you those members that have been contributing. Not to forget those members that have been attending meetings for the last three years. They've given me, and the others that have chosen to stand up in front, the encouragement to keep going.

SUPERTAPE!

From the South Florida ZXUG comes news of a way to link aid to the Ethiopian Relief Fund with our computers. Band-Aid Trust, c/o Stoy Howard, #8 Baker St. London W1, have assembled a Spectrum SUPERTAPE. It includes game programs from Elite, a Horace game from Melbourne House, 3-D

Tank Duel from Real-Time, and six other full scale programs (not forgetting, a "hit song"). "Since all the money goes to feed the starving, this is an incredible offer." Send \$12.95 + tax (& shipping, probably) to the Band-Aid Trust.

An "Orphan Computer?"

News from the local paper indicates that other computer brands may be joining us in orphan status. As of today, Apple fired 25% of their workforce. Sinclair has joined Steve Jobs in being kicked upstairs. I was amused to read an office automation nag blame poor sales on people buying PC's. It looks like, though we thought we were leaving the vital world of computing, we were actually leading the way!

Last Meeting

We had a rip-snorting meeting in June. We started off with balloting for officers, and the verdict was unanimous for the nominated slate. Tony Brooks gave an engaging overview of the Z80 to Microdrive connection, and a (very satisfied) customer's view of Bob Dyl's English Micro Connection. Hank Dickson then stole the show when he introduced two youthful Science Fair contestants - details on that portion can be found further on in the newsletter.

And Next.

First of all, I will be handing the presidency, and leadership of the meetings, over to John Conger. I wish him well, and promise to help him as much as I can. In addition, Wayne Kay will be giving a well-prepared presentation on spike suppression, and power line conditioning in general. Let's all turn out to welcome John and Wayne!

Mark Fisher

TAPE DUBBING

By Jim Mackenzie

The last tape dubbing was a success. I was pleased to see so many tape recorders and participants. Unfortunately some people were unable to make a copy. In the future the tapes will be ten minutes long so that everyone will have a chance. I will also use my ZX 81 to test some tapes. I heard that some people did not get good copies.

I am willing to make a few copies on the low quality tapes I was selling. These I will mail to any eager purchasers for three dollars a tape. Price subject to change without notice. I'll have another tape at the next meeting for the TS 1000. However his will once again simply be more programs from my collection, and i'm tapped out. Please bring some of your programs to me at the meeting or my home. Otherwise there will not be any new tapes to dub. I can be reached at 301-495-9139 after 4 PM, and Mark's number is 301-589-7407. We both live at 700 Erie Ave., Takoma Park, MD 20912.

1985

AD RATES CATS NEWSLETTER

	1X	3X	6X	12X
FULL PAGE	\$100	294	570	1080
HALF PAGE	55	161	313	594
QUARTER PAGE	30	88	171	324
BUSINESS CARD	15	43	81	155

(7" WIDE BY 10" LONG MAXIMUM SIZE. CAMERA READY MATERIAL IN BLACK AND WHITE.)

WE WILL MAIL YOUR ENCLOSURE. READY TO MAIL WITH NEWSLETTER. WE WILL CHARGE ACCORDING TO SIZE OF ENCLOSURE. ASK US FOR QUOTE AND NUMBER NEEDED. (8.5" x 11" SIZE-\$25 PER ISSUE.)

N/L DEADLINE

MEETING DATE

July 19
August 16
September 14

July 13
August 10
September 14
October 12



SUBMISSIONS for this newsletter are eagerly solicited. First priority will be given to member's submissions. Publication of material does not transfer rights from the author, in fact, it may establish priority.

Submissions may be reviews, articles on applications, programming techniques, hardware, or anything else you can imagine. Pertinent articles from other publications will also be considered.

Bring material to the meeting, or send it to PO box 725, Bladensburg, MD 20710. I would prefer material to be typed, single spaced, in 3 1/2" columns - but don't break your back: the Xerox doesn't really care. Printouts from the 2040 printer are fine, but, use Radio Shack paper, and don't put scotch tape over the printing.

Permission is hereby granted for reprints of articles in nonprofit user group newsletters. Please give credit to CATS and the author.

Newsletter Team

- Harvey Altergott
- Tony Brooks
- John Conger
- Monte Copeland
- Hank Dickson
- Mark Fisher
- Sarah Fisher
- Andrea Frankel
- Jules Gesang
- Ed Grey
- London Sunday Times
- Jim Mackenzie
- Ward Seguin
- SFZXUG
- Al Strauss
- Washington Post
- H. E. Wepler

This Month's Puzzle

- Construct a five-line program that will:
- 1.) accept string INPUT (assuming all-alpha string)
 - 2.) PRINT to screen a transposition coded version of the input, transposed a random distance.
 - 3.) PRINT only alphabetic.

Yes, I know you can get it all in one line on the 2068 - single statement lines only!

MF

171 Main St. Apt. 44
Madison, N.J. 07940
June 7, 1985

Mark Fisher
President, CATS

Dear Mr. Fisher,

In the May Newsletter I was impressed by Schrack's program "Precision Multiplication".

I am attaching a revision which shortens that program, reduces the manual inputs to two and adds a decimal fraction capability.

I enjoy the Newsletter and hope to attend a meeting soon.

Keep up the good work.

Sincerely,


H.E. Weppeler

EXPANDED MULTIPLICATION

The short program listed in Figure One will give the complete product of two whole numbers of any length. (Based on Schrack, May, 1985)

The ability to handle decimal fractions can be added to the program by inserting the additional lines listed in Figure Two.

```
10 REM EXPANDED MULTIPLICATION
*****
15 INPUT "FIRST TERM ";A#
20 PRINT "A#" "MULTIPLIED BY:"

25 INPUT "SECOND TERM ";B#
30 PRINT B# "EQUALS:"
75 DIM P$(LEN A#+LEN B#+1)
80 FOR I=LEN A# TO 1 STEP -1
85 LET C=0
90 FOR J=LEN B# TO 1 STEP -1
95 IF P$(I+J)=" " THEN LET P$(I+J)="0"
100 LET Q=VAL A$(I)*VAL B$(J)+VAL P$(I+J)+C
105 LET C=INT (Q/10)
110 LET P$(I+J)=STR$ (Q-10*C)
115 NEXT J
120 LET P$(I)=STR$ C
125 NEXT I
135 IF P$(1)="0" AND P$(2)<>" " THEN LET P$=P$(2 TO ): GO TO 15
160 PRINT P#
165 STOP
```

FIGURE ONE

MACHINE CODE NOTES

Our Machine Code classes will continue to meet on the first and third Saturdays of July and August at 10 am to 1 pm (summer hours) at the Chevy Chase Library. To cover the text material to the point where we should become reasonably competent will take about twelve sessions, or to mid November. The text is Rodney Zaks' Programming the Z80, available at Maryland Book Exchange.

If there is enough demand for a weekday evening class, we will set one up in a convenient location. Give me a call in the next few days if evening classes are better for you.

I have written some lecture notes and diagrams for class use which will be distributed to class members. Some of these are being adapted for presentation in the Newsletter. When they are all assembled over the next six months they should make a simplified Handbook on Machine Code to use as a refresher.

We have held two MC classes to date (June 13) with a third class scheduled for June 15th. So far we have covered:

- Organization of the Z80 chip
- Binary and Hexadecimal number systems
- Binary addition and subtraction
- Introduction to the Assembler
- Writing a machine code routine to multiply one byte numbers

June 15th we are scheduled to cover improving the multiplication routine, using the Hot Z Assembler, and also we will cover Computer Logic with the aid of Brian Little.

It is still not too late to sign up for the weekday or Saturday class, so give me a call if you are interested. (John Conger 654-5751)

```
40 FOR D=LEN A# TO 1 STEP -1
45 IF A$(D)="." THEN LET A#=A#
( TO D-1)+A$(D+1 TO ): LET D=LEN
A#-D+1: GO TO 55
50 NEXT D
55 FOR E=LEN B# TO 1 STEP -1
60 IF B$(E)="." THEN LET B#=B#
( TO E-1)+B$(E+1 TO ): LET E=LEN
B#-E+1: GO TO 75
65 NEXT E
135 LET I=I+1: IF I>(D+E) THEN
GO TO 155
140 IF P$(LEN P#-I)="0" THEN LET
P$(LEN P#-I)=" ": GO TO 135
145 LET P#=P$( ( TO LEN P#-D-E-1)
+" "+P$(LEN P#-D-E TO )
```

FIGURE TWO

2K CALCULATOR

ZX-T/S 1000

by Monte Copeland, South Florida ZXUG
Boynton Beach, FL 33425-0951

Here's an idea for you 2K TS1000 users. This program turns your 1000/2040 system into a printing calculator. This is also a very educational program, in that it uses the Hewlett-Packard type of Reverse Polish Notation. Any of you who are learning FORTH, take note; this may be just the sort of practice you need.

COMMANDS

"D"	display stack
"C"	clear stack
"P"	print top of stack
<ENTER>	push # on stack
"K"	add
"J"	subtract
"B"	multiply
"V"	divide

To enter a number you press ENTER, then type the number or expression and press ENTER again. The structure of the program is; a case statement within an infinite loop. After you key in the program, SAVE it by typing GOTO 3000. This will make the program "auto-starting."

This program may not be the most practical way to get a printout of your calculations. It does, however, demonstrate some great features. It shows RPN, the use of a push-down stack, and how to program a case statement from BASIC.

```
10 DIM S(50)
12 LET SP=0
15 GO TO 2E3
40 IF INKEY$="" THEN GO TO 40
50 LET I=CODE INKEY$
60 LET BSP=SP-1
100 IF I<>110 THEN GO TO 200
110 INPUT X
120 LPRINT X
130 LET SP=SP+1
140 LET S(SP)=X
150 GO TO 2E3
200 IF I<>61 THEN GO TO 300
220 LET SP=0
225 LPRINT "STACK CLEAR"
230 GO TO 2E3
240 NEXT N
300 IF I<>41 THEN GO TO 400
320 FOR N=SP TO 1 STEP -1
330 LPRINT N; " "; S(N)
360 GO TO 2E3
400 IF I<>53 THEN GO TO 500
410 LPRINT "STACK TOP = "; S(SP)
420 GO TO 2E3
500 IF I<>48 THEN GO TO 600
510 LET S(BSP)=S(BSP)+S(SP)
520 LET SP=BSP
```

Puzzle Solutions!

Here are two member's solutions to last month's puzzle. Though none of them does the job in exactly the way I had in mind, each of the four uses some unusual programming trick (OK, algorithm) to get the job done.

From the Barasches:

```
10 FOR A=1 TO 3
20 FOR B=1 TO 3
30 PRINT A;",";
40 NEXT B
50 NEXT A
(Nested loops)

10 INPUT A$
20 DIM B$(LEN A$)
30 FOR K=1 TO LEN A$
40 LET B$(K)=A$(J-K+1)
50 NEXT K
60 PRINT B$,A$
(complex string slicing)
```

From Al Strauss:

```
10 LET A$="X"
20 LET X=1
30 FOR N=1 TO 3
40 PRINT VAL A$(1)+A$(2)
50 NEXT N
40 LET X=X+1
40 IF X=11 THEN STOP
50 GOTO 25
```

(Use of VAL with string slicing)

```
10 LET A$="ABLE WAS I ERE I SA
W ELBA"
15 LET X=LEN A$
20 FOR N=X TO 1 STEP -1
30 PRINT A$(N)
40 NEXT N
```

(Loop with STEP -1)

```
525 LPRINT "+"
530 GO TO 2E3
600 IF I<>47 THEN GO TO 700
610 LET S(BSP)=S(BSP)-S(SP)
620 LET SP=BSP
625 LPRINT "-"
630 GO TO 2E3
700 IF I<>39 THEN GO TO 800
710 LET S(BSP)=S(BSP)*S(SP)
720 LET SP=BSP
725 LPRINT "*"
730 GO TO 2E3
800 IF I<>59 THEN GO TO 900
810 LET S(BSP)=S(BSP)/S(SP)
820 LET SP=BSP
825 LPRINT "/"
830 GO TO 2E3
900 IF I<>38 THEN GO TO 2E3
910 REM SLOW
930 CLEAR
940 STOP
2000 IF INKEY$="" THEN GO TO 2E3
2010 GO TO 40
3000 SAVE "C"
3010 LPRINT "OK"
3020 GO TO 0
```

CRITICAL PATH METHOD (CPM)

This 2068 program calculates the minimum amount of time required to perform a set of related activities and identifies the "critical" path. The critical path is that set of activities that together determine the minimum amount of time required.

Each activity begins and ends with a "node" or milestone, as in a PERT chart. For example, if activity 1 was to go from Washington to Chicago, the leaving of Washington would be the start node which could be identified by any number, and the arrival in Chicago would be the end node which would also be identified by a number.

As an example, getting to work could be considered to be five activities: 1 getting dressed, 2 getting a shower, 3 shaving, 4 getting breakfast at restaurant, and 5 taking subway; as shown in the table.

activity	start	end	time	cost
1 shave	1	2	10	0
2 shower	1	2	15	0
3 dress	2	3	19	0
4 eat	3	4	44	4.25
5 ride	4	5	55	2.75

In this case activities 2,3,4, & 5 form the critical path of 132 minutes and total cost is \$7.00.

The program can handle 100 different activities. This can be increased by changing line 70.

Harvey Allright

PLEASE! Use Radio Shack thermal paper, and BOLD CHARACTERS, as shown in the June and April newsletters. Ed.

```

10 REM CPM
11 REM TS2068 - to calculate c
critical path of network
12 REM events begin & end at "
nodes"
20 REM A#=START AND END NODES
FOR EACH ACTIVITY
30 REM S#=EARLY START TIMES FO
R EACH ACTIVITY
40 REM F#=LATE FINISH TIMES FO
R EACH ACTIVITY
50 REM E#=DURATIONS AND COSTS
OF NORMAL ACTIVITIES
60 REM C#=DURATIONS AND COSTS
OF CRASH ACTIVITIES
70 DIM A(100,2)
71 DIM S(100)
72 DIM F(100)
73 DIM E(100,2)
74 DIM C(100,2)
90 PRINT "CRITICAL PATH METHOD

100 PRINT
110 PRINT "HOW MANY ACTIVITIES"
"
111 PRINT "IN THIS NETWORK?"
120 INPUT N
130 FOR I=1 TO N
140 PRINT
150 PRINT "ENTER DATA FOR EACH
EVENT"
151 PRINT "EVENT START END
TIME COST"
152 FOR I=1 TO N
153 PRINT I:
154 INPUT A(I,1)
155 PRINT " ";A(I,1);
156 INPUT A(I,2)
157 PRINT " ";A(I,2);
170 IF A(I,2)<=A(I,1) THEN GO T
O 200
190 IF A(I,2)<(N+2) THEN GO TO
260
200 PRINT "START NODE MUST BE N
UMBERED LOWER THAN END NODE, AND
END NODE CAN NOT EXCEED THE NUMB
ER ACTIVITIES BY MORE THAN ONE."
230 PRINT " ***TRY AGAIN***"
240 PRINT
250 GO TO 153
270 INPUT E(I,1)
271 PRINT " ";E(I,1);
272 INPUT E(I,2)
273 PRINT " ";E(I,2)
280 LET S(I)=0
290 LET F(I)=0
300 NEXT I
310 REM LOOP TO FIND EARLY STAR
T TIMES FOR NETWORK
320 FOR I=1 TO N
330 IF S(A(I,2))>S(A(I,1))+E(I
,1) THEN GO TO 350
340 LET S(A(I,2))=S(A(I,1))+E(I
,1)
350 NEXT I
360 LET F(A(N,2))=S(A(N,2))
370 REM LOOP TO CALCULATE LATE
FINISH TIMES FOR NETWORK
380 FOR I=N TO 1 STEP -1
390 IF F(A(I,1))=0 THEN GO TO 4
20
400 IF F(A(I,1))>F(A(I,2))-E(I
,1) THEN GO TO 420
410 GO TO 430
420 LET F(A(I,1))=F(A(I,2))-E(I
,1)

```

Creating a Back-up Cassette for the 2068 Program UU-FILE

If you have ever attempted to make a back-up copy of some of the TS software such as "UU-FILE" or "UU-CALC" using the SAVE program function, you soon found out it doesn't work. The reason, the bulk of the program is in machine code. But it can be done and Appendix E of the 2068 manual gives a clue as to how to do it. It is not obvious so let me tell you what I've learned.

First I'll list the steps and then explain a couple of points. Even if you don't understand my explanation, you should be able to duplicate my steps.

1. Type in: MERGE "vfile"
2. Start the UU-FILE tape and press ENTER
3. Stop the tape as soon as you see OK on the screen.
4. Display the BASIC program by pressing: ENTER

```
50 BORDER 1: PAPER 1: INK 1: C
LEAR 28287: LOAD ""SCREEN$: LOA
D ""CODE
100 INK 7: DIM f$(32): LET a=28
291: GO TO USR 28288
1000 CLS : PRINT "Save a file":
GO SUB 6000: GO TO USR a
1005 SAVE a#CODE s,l: PRINT "Pre
ss V to VERIFY "" or any other k
ey to continue."
1010 IF INKEY#="" THEN GO TO 101
0
1020 IF INKEY#(<"V" AND INKEY#(<
"V" THEN GO TO 1200
1100 GO SUB 7000: VERIFY a#CODE
1200 GO TO USR a
2000 CLS : PRINT "Load a file":
GO SUB 6000: GO SUB 7000: LOAD a
#CODE s: GO TO USR a
3000 LPRINT f$( TO 32): GO TO US
R a
6000 INPUT "Enter file name: ":
LINE g$: LET f$( TO 10)=g$: LET
s=CODE f$(11)+256*CODE f$(12): L
ET l=CODE f$(13)+256*CODE f$(14)
: RETURN
7000 PRINT ""If the program stop
s executing and returns to BASI
C with an error code then ent
er:"""GO TO USR a"" to get bac
k into UU-FILE."": RETURN
```

5. Edit line 50 by deleting LOAD "SCREEN":
6. Start the cassette again and enter: RUN ENTER

7. Stop the tape when the UU-FILE menu appears on the screen.
8. Now save the program on a blank cassette. Type in: SAVE "vfile" LINE 50
9. As soon as OK appears stop the tape.
10. Type in: SAVE "xxxx" CODE 28288, 37080
11. Hit: ENTER and start the tape
12. Stop the tape when OK appears. The program file should be on your back-up cassette. Delete the program from memory and try reading it off your back-up tape.

There are three modules in the UU-FILE (and likewise in UU-CALC) program. The first is a basic program module which initializes some of the variables; but most important, it instructs the machine to read in the next two machine code modules. The first is the screen display which you see when the second machine code module is being loaded. It's neat but you don't need the screen display for tape back-up purposes. Therefore, step 5 above deletes the screen display module. The second machine code module is the functional part of the program that performs what the program is advertised to do.

At the completion of Step 7, you have the entire program in the machine except for the screen display. Steps 8-11 save the program on your back-up tape, first the basic code and then the machine code. Step 8 saves the program so that the basic code automatically starts with line 10 (for UU-FILE) when the program is loaded from the back-up tape.

Step 10 saves the machine code, all 37080 bytes, starting with address 28288. The machine code module may not be 37080 bytes long but lacking knowledge of how big the program is, I calculated this number. I knew RAMTOP is normally 65367 (see page 268 of the manual) and I knew the basic program loaded the machine code starting with address 28288 because statement 50 of the basic program says CLEAR 28287. The difference between these numbers is 37080.

SCIENCE FAIR EXHIBITORS FIND SINCLAIR COMPUTERS
A BIG HELP IN FURTHERING ROBOTICS AND CONTROLS

The 1985 Prince George's Area Science Fair was held last April on the Largo campus of P.G. College. It brought together 300 outstanding students from 50 high schools located in four southern Maryland counties. Their projects were distributed over twelve scientific categories, ranging from biochemistry to zoology.

Two of the exhibitors employed Sinclair or TIMEX/Sinclair computers in carrying out their projects. This proved to be both a useful and successful technique, since it helped them earn the honors and recognition in their local fairs that made them eligible to represent their schools at the Area Fair.

These two P.G. county students found time in their busy end-of-school-year schedules to appear before an attentive and appreciative June CATS meeting to display first-hand their slick computer-oriented science fair projects.

RICK COVELL

Senior

Oxon Hill High School

"Robotics via Computer Control"

Rick has demonstrated a keen interest in robotics for several years. Last year he was the grand award prize winner among the 400 entries in the Oxon Hill science fair using a radio-controlled robot he had constructed.

In June, Rick graduated with the first full four-year class of students to wend its way through the Oxon Hill H.S. Science and Technology Program. This is a highly competitive program now offered as part of the curriculum within two P.G. County high schools. It prepares the students for further education in high technology subjects. Although a highly academic program, Rick still found his favorite courses were his shops electives.

One of the graduation requirements for the Science and Technology program is a "Research Practicum". It is based on a year-long research or engineering project. Rick's robotics project was the visible evidence of his research practicum. It was supported by a thoroughgoing written research document.

The project Rick displayed to CATS consisted of a robotic manipulator arm under the computer command of a T/S 1000. This prototype proved to have the same control problems encountered with a light duty industrial arm.

Creating it involved understanding of electricity, electronics, microprocessors, microcomputers, instrumentation and control, and mechanics.

After all the hurdles were cleared in designing and building the mechanical arm, Rick was able to locate in a magazine for industrial arts teachers an 18-line BASIC program to provide general control for the arm. A ByteBack BB-1 interface functioned between the Sinclair and the relays that drove the stepping motors and lead screws on the mechanical arm.

MIKE O'NEILL

Ninth Grade

Bowie High School

"Computer Control of Outside Devices"

Mike has a ZX81 which he and his father put together some time ago. He has added 64Kb of memory to it. A couple of years ago he started to become interested in ways the ZX81 could be used to control various external devices.

He proceeded to fashion his own custom interface to control such devices, using the port at the back of his Sinclair. (See accompanying diagram.)

For his science fair project, he hooked up a display with four devices:

A light bulb

An AM radio

A beeper

A fan motor

The interface he made can handle up to eight devices---one for each bit on an eight-bit bus.

He created his own BASIC program to permit easy operation by the user. The program has a sparkling menu-type screen. There is the ability to use an attached joystick to select the desired menu choice and thereby start or stop the desired external device.

Between the interface at the back of the Sinclair and the devices is a group of relay look-alikes which translate the on/off signals from the computer into action. These are actually "open collector transistors" or UHP-400 chips, which function in a manner similar to relays.

Taken all together, Mike's project demonstrates a positive way of elevating small computers such as the Sinclair into a realm where they can do useful work.

For the future, Mike has plans to inject some inputs into his Sinclair using items such as heat sensors. These inputs will then be used to trigger functions such as watering systems to provide moisture for plants or cooling for dwellings.

Rick borrowed a T/S 1000 from a friend to drive the arm for the science fair presentations. He never bothered entering the 18-line program from a cassette--he just entered it fresh every time he set up a demonstration. For one thing, it seemed the T/S 1000 was terminally unable to load from tape. So Rick was just glad that he didn't have to contend with a 180-line program!!

(Incidentally, CATS member Sam Lefkov has generously offered to replace Rick's defective T/S 1000 with one of his own!)

This fall, Rick will be entering the Florida Institute of Technology in Melbourne, Florida, where he will continue his studies in robotics.

The mechanical arm, construction costs of which were partially underwritten by the school, will remain behind as a teaching tool for future generations of roboticists at Oxon Hill High School.

Mike's favorite subject in school is mathematics. His project demonstrates the application of good mathematical concepts to a series of practical applications.

For future Area Science Fairs, the mathematics and computer projects may be separated, which would definitely be to Mike's advantage. He would then be competing with other computer projects exclusively, rather than against the pure mathematics projects as well as the computers.

We look forward to seeing Mike's work as it progresses through the years to come!!

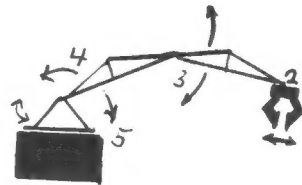
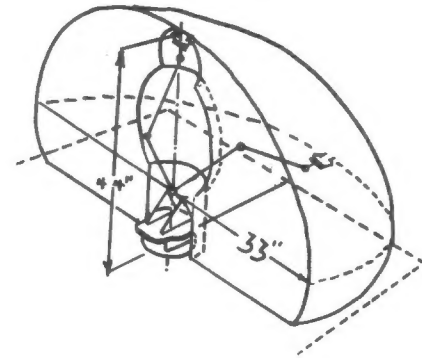
What's Ahead

RICK COVELL will be taking his interest in robotics and computer control for robots to the Florida Institute of Technology this fall.

MIKE O'NEILL will be pursuing his computer-drive process controls during the next school year, with certain modifications and enhancements to this year's project already taking place in his mind.

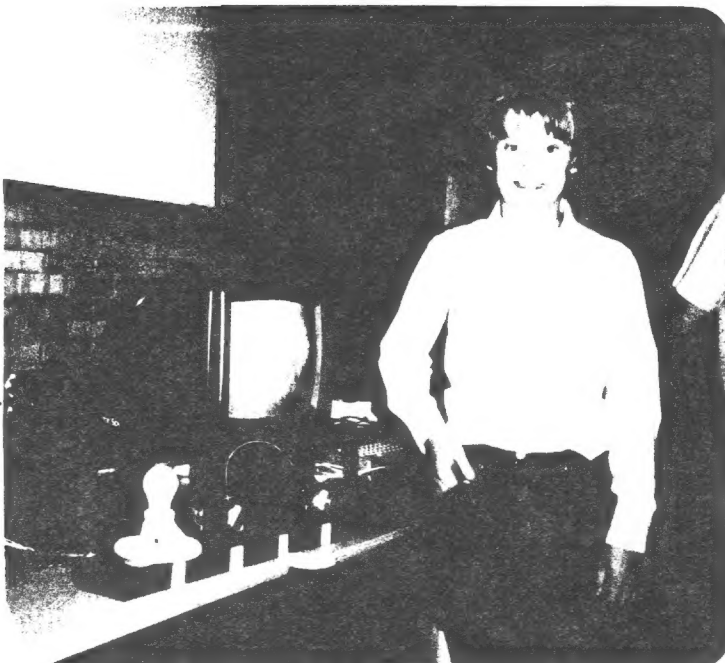
In recognition of their fine work and their appreciation of the usefulness of small computers, both young men were made honorary members of CATS by acclimation at the closing of their presentations.

Volume Diagram

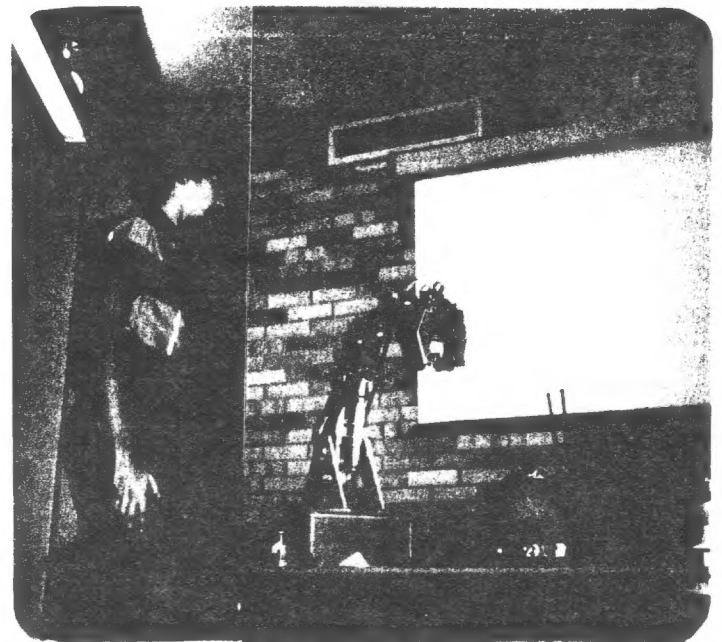


Basic Functions

- 1 hand / gripper
- 2 wrist
- 3 elbow
- 4 shoulder
- 5 base



Mike O'Neill



Rick Covell

BYTE-BACK

IN THE BEGINNING THERE WAS SINCLAIR
THEN THERE WAS BYTE-BACK

INC.
RT. 4 BOX 54, LEESVILLE, SC 29078
BYTE-BACK HAS BEEN MANUFACTURING PERIPHERALS FOR TIMEX / SINCLAIR COMPUTER OWNERS SINCE 1981. IT HAS ADDED MANY NEW PRODUCTS SINCE ITS FIRST 814 AND 88-1 CONTROLLER. BYTE-BACK HAS PULLED THROUGH THE DENISE OF TIMEX AND WILL CONTINUE TO PROVIDE SUPPORT AND NEW PRODUCTS IN THE FUTURE.

MODEM

19-28 for ZX81, TS1000, TS1500
19-48 for TS2048
upload and download test to and from memory and tape
300 baud - word length, parity, etc. selectable
direct connect with phone load for clear communication

ALL THE FEATURES NECESSARY FOR TELECOMMUNICATIONS WITH SERVICES SUCH AS COMPUERVE OR YOUR LOCAL BBS
RS-232 port for adding full size printers
2 hour comserve demo pack
5 hour comserve starter kit available separately

64K MEMORY

19-44 for ZX81, TS1000
NOT AVAILABLE FOR TS2048
8-16K area selectable in 2K increments
eprom socket for programmed eproms (2K and 4K)
reset switch which only resets the 16 to 20K area

EXPAND YOUR TIMEX/SINCLAIR TO ITS FULLEST POTENTIAL
battery back up to stop annoying crashes
rom-transfer for machine code buffs
this is not a nonvolatile ram

CONTROLLER

88-1 for ZX81, TS1000, TS1500
88-48 for TS2048
8 independent relays
8 led status indicators to monitor status of relays
analog to digital converter sold separately to allow controller to measure voltages

CONTROL THINGS WITH YOUR TIMEX/SINCLAIR
8 logic inputs
YOUR TIMEX SINCLAIR CAN OPERATE FULL SIZE RS-232 TYPE PRINTERS
C.ITON 7306AP serial printer available separately

RS-232

RS-232 for ZX81, TS1000, TS1500
RS-232-48 for TS2048
cable and software sold separately

YOUR TIMEX SINCLAIR CAN OPERATE FULL SIZE RS-232 TYPE PRINTERS
C.ITON 7306AP serial printer available separately

PARALLEL

PARALLEL 1600 for ZX81, TS1000, TS1500
PARALLEL 2048 for TS2048
cable and software provided

YOUR TIMEX SINCLAIR CAN OPERATE FULL SIZE CENTRONICS PARALLEL TYPE PRINTERS
C.ITON 7306AP parallel printer available separately

90 DAY WARRANTY ON ALL MODULES

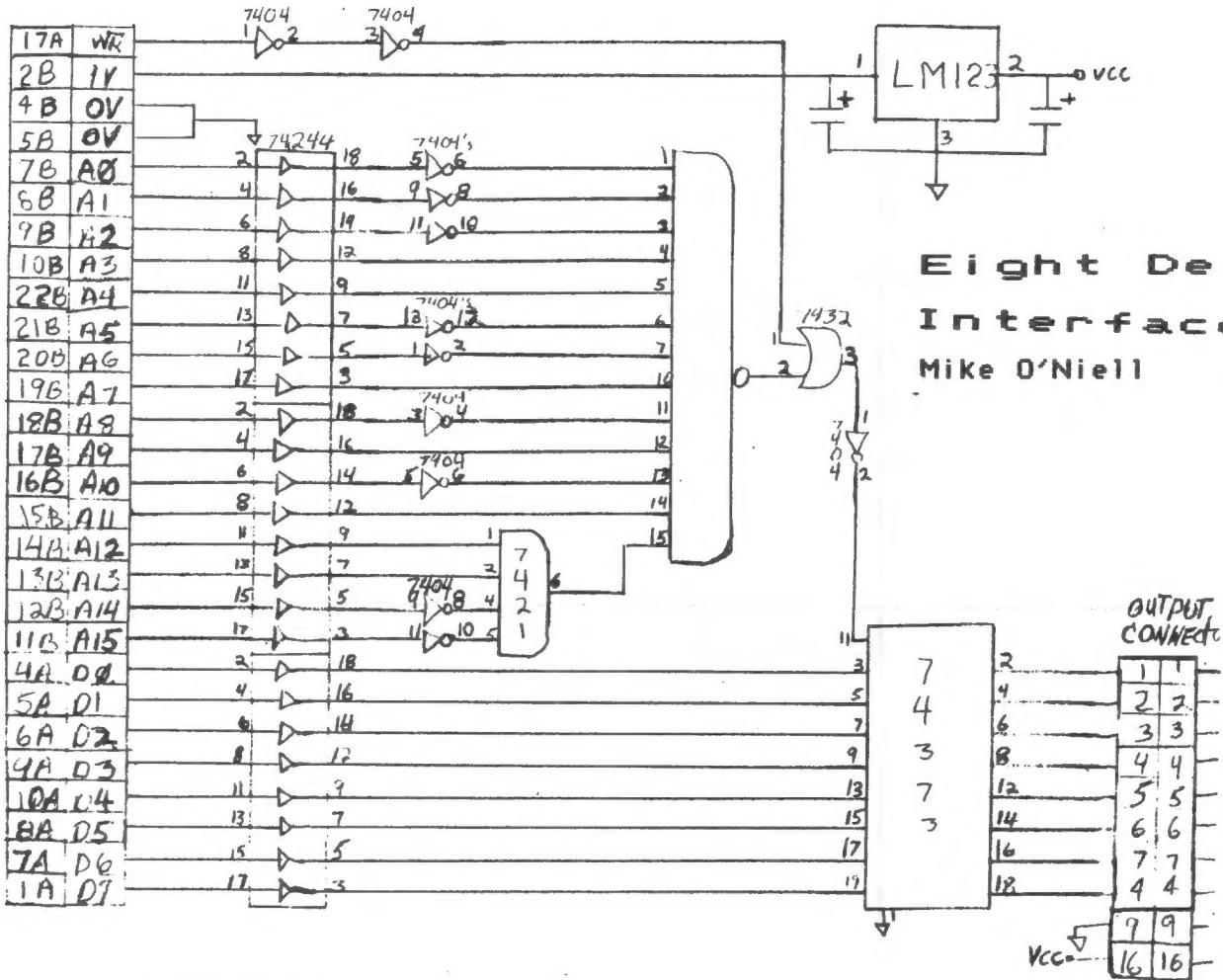
Any hardware module may be returned for a full refund within 90 days of receipt.
BYTE-BACK offers a few hardware kits for kit builders. Our kits require excellent soldering ability.
Kits include a high quality printed circuit board and all of the parts.
Kits have a 90 day warranty on parts only. Kits may not be returned for a refund once they have been started.

TELEPHONE (803) 532-5812

19-44 assembled \$119.95
19-44 assembled \$ 67.00
19-48 assembled \$ 69.00
19-232 assembled \$ 49.95
19-232-48 assembled \$ 69.95
19-28 assembled \$149.95
19-48 assembled \$149.95
A-D converter assembled \$ 29.95
Test lead for A-D \$ 3.95
PARALLEL 1600 assembled \$ 84.95
PARALLEL 2048 assembled \$ 84.95
CABLE FOR RS-232 \$ 19.95
PRINTER SOFTWARE FOR RS-232 \$ 19.95
ZX PRO/FILE (TS1000) \$ 14.95
ZX PRO/FILE (TS2048) \$ 29.95
COMPUERVE 5 HOUR PACKAGE \$ 29.95
7306AP SERIAL PRINTER \$327.00
7306AP PARALLEL PRINTER \$299.95

111 \$189.95
111 \$ 39.00
111 \$ 39.00
111 \$ 39.95
111 \$ 39.95
CARD #
EXP DATE
NAME
ADDRESS
CITY/STATE/ZIP
PHONE

THERE WILL BE A \$4.95 SHIPPING CHARGE PER ORDER.
PLEASE SPECIFY COMPUTER TYPE
PLEASE BILL MY AM. EXP. - VISA - M/C
I HAVE ENCLOSED A CHECK OR MONEY ORDER



Poor Mans Telephone Modem

Poor Man's MODEM

by Anon., South Florida ZXUG
Boynton Beach, FL 33425-0951

Now you can send and receive programs by telephone, without a MODEM. In fact, all you need is a small transformer, an on/off switch, and an earphone jack - total price, about \$10.00.

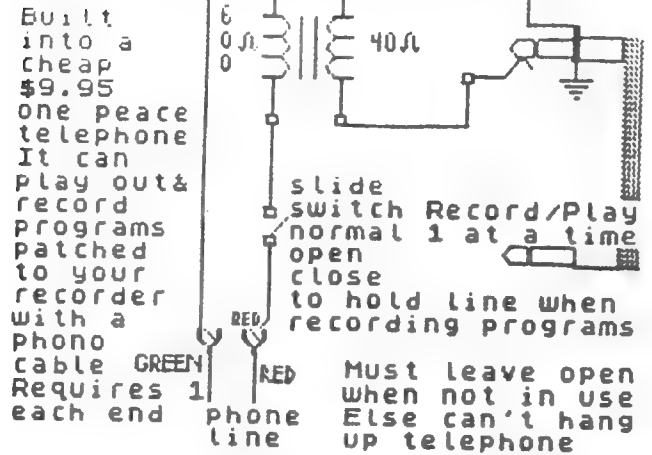
The transformer must have about 500-600 ohms on the side that connects to the phone line, and about 30-100 ohms on the computer side (DC Ohms as read on any ohmmeter). I have found that the intermediate transformer in a solid state audio amp will usually work well as long as the resistance is as stated. The wiring is simple. The easiest way to do it is to build this into a cheap phone. If you don't have one, you will either have to buy one, or get a phone splitter such as Radio Shack #279-357 or #270-373 and a phone cord that has a modular plug on one end and wires on the other (#279-391 or #279-364). If you have the older type jacks that are wired to the wall, you may wire it in, but it should be able to be removed.

First connect the GREEN wire from the phone cord to one lead of the 600 ohm side of the transformer. Now connect the RED wire to one wire of the switch. The other wire of the switch goes to the remaining wire of the 600 ohm side of the transformer. Now all that is left is to connect the other two leads of the transformer to the earphone jack, or to a shielded cable terminating in an earphone jack. You can put it in a small box or build it into a cheap phone (which will avoid the purchase of a phone cord).

Using it is also simple. To send a program, set your volume on your recorder to about 1/4. Plug the cable into the phone and the earphone to the transformer. This is assuming that you have already called your friend. Now have him (her) plug their adapter into the phone and their MIC jack. Now have them start their recorder in RECORD mode and cover the phone's own mike, or hold MUTE on. Now play the program to them just as if you were loading your computer! After you have sent it, Have them try to load it. If it won't load try a different volume level. I have even plugged the transformer into a cheap audio amp (Radio Shack #277-1008 @ \$11.95) and loaded the program right into the computer!

It will take some trial and error to find the right volume levels but if it saves some driving I think it's worth it!

[Ed. note: I don't know what Ma Bell would think about this project, but I guess she's got other things on her mind right now. Just don't fry the central office!]



```

430 NEXT I
440 LET C1=0
450 LET L=0
460 LPRINT
470 REM CALCULATE BLOCK TIME IN
480 LPRINT "START END EAR
L: LATE"
490 LPRINT "NODE NODE STA
AT FINISH DURATION BLOC
COST"
500 FOR I=1 TO N
501 LPRINT
510 LPRINT A(I,1);TAB(19);A(I,2)
;TAB(17);B(A(I,1));TAB(23);
520 LPRINT F(A(I,2));TAB(33);E
(I,1);TAB(45);
530 LET S1=F(A(I,2))-B(A(I,1))-
E(I,1)
540 IF S1<0 THEN GO TO 590
550 LPRINT "CRITICAL"
560 LET L=L+B(I,1)
570 GO TO 500
580 LPRINT
590 LPRINT TAB(56);E(I,2)
600 LET C1=C1+E(I,2)
610 NEXT I
620 LPRINT
630 LPRINT "THE CRITICAL PATH L
ENGTH IS "L
640 LPRINT "TOTAL COST OF THIS
NETWORK="C1
650 LPRINT
660 PRINT "DO YOU WANT TO CHANG
E DURATION OR COST OF ANY EVENT?"
670 INPUT A#
680 IF A#="N" THEN GO TO 670
690 IF A#<>"Y" THEN GO TO 660
700 PRINT
710 PRINT "WHICH ACTIVITY?"
720 INPUT I
730 IF I<1 THEN GO TO 720
740 IF I>N THEN GO TO 720
750 PRINT "CURRENT DURATION OF
E(I,1)="E(I,1);" COST ="E(I,
E)
760 PRINT "ENTER NEW DURATION A
ND COST"
770 INPUT E(I,1);E(I,2)
780 PRINT "---RECALCULATING NET
WORK---"
790 LPRINT
800 FOR I=1 TO N
810 LET S(I)=0
820 LET F(I)=0
830 NEXT I
840 GO TO 510
850 STOP

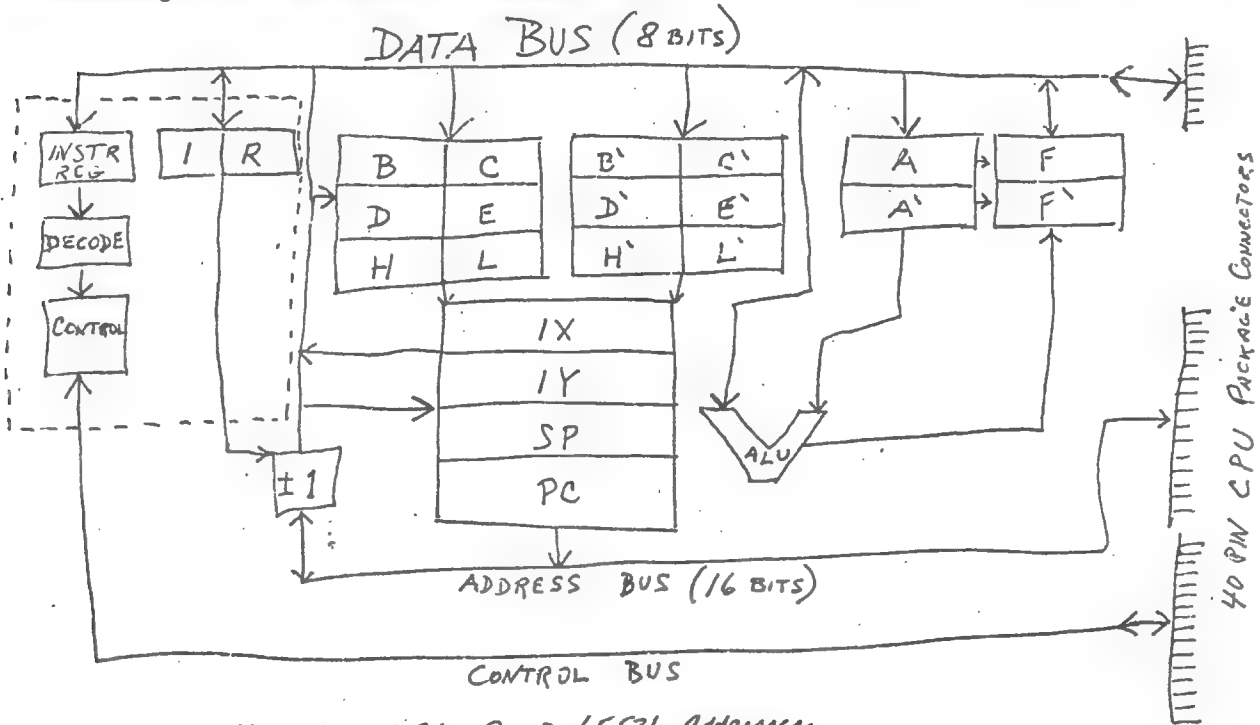
```

280 Chip Organization and 64K Memory Diagram

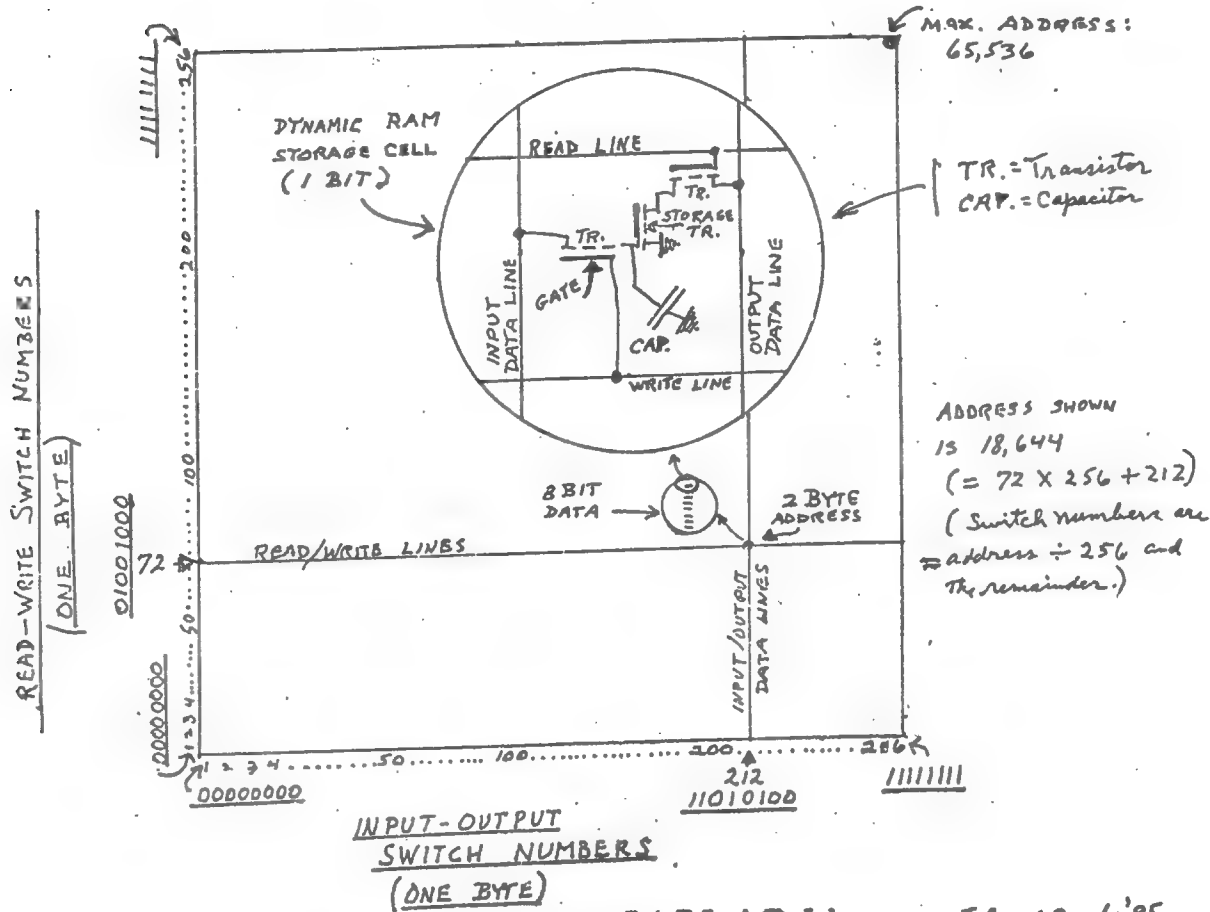
In the Machine Code class, the two diagrams below were used to introduce the concepts of 280 memory registers and RAM addresses necessary to an understanding of how and where data and data

addresses are handled by machine code programming.

If you want them fully explained, come to the next MC and Assembly Language class. (See article: "Machine Code Notes".) (John Conger 654-5751)



How 16 CPU Pins Reach 65,536 Addresses



MSCRIPt Tips

MSCRIPt is a great program. The documentation, however, isn't so great. MSCRIPt was written to apply to a variety of machines, and the documentation provides only one short page of specific instructions for the 2068. In this article, I'll discuss two major classes of omissions in the documentation - missing commands and detailed SAVE and LOAD information. As a conclusion, I'll cover some points that can make using MSCRIPt even more useful.

The Missing Commands

Both the manual and the HELP screen ignore several important commands. These commands relate to cursor movement, and all are reached through the caps + symbol shift, or Function key. These commands should have been included in the HELP screen, but they weren't.

Fn-5: This brings the cursor to the left margin in one stroke, similar to a carriage return on a typewriter, but without the newline.

Fn-8: This is TAB. It is referred to in the documentation as the TAB key, but its location on the 2068 isn't mentioned. The cursor will move to the next TAB position as shown on the status line at the bottom of the page. TAB positions are changed through the Main Menu. (To find the TAB values you will need, move the cursor to the position needed, and note down the Column position from the bottom line of the screen.) (As mentioned in the May Newsletter, TAB can be moved to CAPS-shift 1 by POKEing 42661,152 before entering MSCRIPt.)

Fn-6: Page down. This leaves the cursor in the same screen position, but writes the next 22 lines of text to the screen.

Fn-7: Page up. This moves the text on screen up 22 lines.

SAVE and LOAD Info

MSCRIPt is completely self-contained: unlike almost all other programs, it makes no use of the original Timex operating system. In order to read or write to a cassette, then, MSCRIPt contains its own code. In writing this code, the authors didn't follow the Timex system completely.

SAVEing

The Save command is reached by entering S or s as the first character in the command line. The Timex requires quote marks around the SAVE name.

MSCRIPt, however requires no quotes. A text file can be SAVEd under the name "letter" by typing:
s letter

MSCRIPt will ignore leading spaces; "s letter" or "sletter" will SAVE as the same. Once the command line is ENTERed, the familiar phrase "Start tape and press ENTER" will appear. As with the Timex SAVE, data will be put out as soon as ENTER is pressed that second time. Unlike the Timex, MSCRIPt does not echo the tape signal to the screen, so the only way to tell if the signal is finished is that the text reappears on the screen.

In addition, the data format that MSCRIPt uses is different than Timex's, and MSCRIPt files cannot be loaded by the standard Timex load.

LOADing

LOADing is different, as well. If your file was saved as "letter," as in the example above, it can be reloaded by using: l letter -or- ll letter

-or- ll

Using "l letter" will bomb with the message "Tape IO Error!". Once the command line is ENTERed, you will again see the "Start tape and press ENTER" prompt. MSCRIPt means it! Press ENTER again, and MSCRIPt will listen for an incoming signal from the tape. Again, MSCRIPt does not echo the signal to the screen, so it can be difficult to tell where to start. In addition, MSCRIPt won't tell what text files it's reading - it either loads the file, or prints "Tape IO error!".

To counteract the lack of a visual echo to the screen, I recommend adding some sort of monitor to the tape recorder - either an LED that shows when a program is playing, or a high-resistance bypass to the speaker, allowing a faint echo to be heard.

To compensate for the lack of the tape's file name on screen, the best defense is an accurate log of each cassette's contents. If that fails and you have forgotten the name the file was SAVEd under, you have two choices: either LOAD using the first letter of the file (if you remember that), or use the HEADER program from the library tape to read the leader. Although the format of the header is different, HEADER will read the second through last character in the file name. HEADER won't tell you that critical first letter, but it should jog your memory - at least it's better than trying all upper and lower case letters.

VERIFY=APPEND

Alas, VERIFY doesn't exist in MSCRIPt. However, MSCRIPt does include an important extra command, APPEND. The syntax is the same as other MSCRIPt tape commands, and it will non-destructively allow you to combine an existing document with another one from

the tape. This can be used as a form of VERIFY for text files that are less than half the maximum. After the file is SAVED, just rewind the tape, and APPEND the file again. If it was a good SAVE, it should go in easily, leaving you with two copies of the text file in memory. Check for obvious glitches, then block delete the second copy.

As a footnote, it should be noted that these problems of MSCRIPT's version of tape IO may evaporate soon. In order to use MSCRIPT on disc, it is necessary to exit MSCRIPT. Versions that allow this have already been created by several ace programmers, including Jack Dohaney. Once MSCRIPT is exited, standard LOAD and SAVE commands can work.

Everyday Use: Text Formatting

As you get accustomed to MSCRIPT, you will learn to use the special features offered by your printer. MSCRIPT allows these, but requires some homework in setting up the definitions of its special characters at some time before they appear in the text. If your writing tends to follow the same format, you may save time by creating a text file that contains only formatting information - line length, margins, special printer definitions, etc. I went beyond this, and created a blank letter form, with spacing already worked out. When doing a series of reports, I created a report form and SAVED it, thus speeding up the actual report writing, as I was constantly prompted for the next item to include. If you're using special TAB settings, you won't be able to save or load them; instead, just insert a comment line documenting the TAB positions you need, and insert them when you restart the program.

Multiple Copies

MSCRIPT has no provision for printing multiple copies, but you can persuade it to print a few, if your document isn't too long. When the text is completed, enclose the entire text file with block markers as the first and last characters, move the cursor just beyond the last block marker, and press Fn-C for as many copies as you want (or have space for). Print as normal.

Printing Portions of Text.

MSCRIPT can't "officially" exclude a portion of text from being printed. A ">*" at the head of a line keeps that line from printing, but it's not effective for large blocks of text. However, there are two strategies you can use to exclude large blocks of text: 1) SAVE the full text to tape, and block-delete the unneeded portion before printing. 2) Block-move the desired text to the head of the

file, and end it with a new-page marker. Print in Single-Sheet mode, and when the text you want to print is on paper, press break (this also works in Memotext).

Command Syntax

MSCRIPT is very tolerant of variations of syntax. The manual specifies commands such as "JU=Y" in its discussion, but MSCRIPT will treat "ju=y", or even "juy", as the same command. The full syntax is easier to read when you're trying to figure out why your printer just shot its platen through the wall, but the abbreviated syntax is quicker to type.

Don't forget to keep an accurate log of text files, and think about adding an audible monitor to the tape player. Lastly, if a text file doesn't go in, press ENTER again, rewind the tape, and try again - you probably forgot to press ENTER twice. Good luck!

HF

Facemaker for telecom

I recently joined the ranks of UNIX-users, and started reading the collection of electronic bulletin boards who are known collectively (with their users) as "Usenet." . . . A bit of "netiquette" that I found amusing was the convention of using "SMILEYS." These are :-) :-) B-) :@) 8-) etc. (If you turn your head 90 degrees, they look like smiling faces. #1 is winking, #2 is the "standard" smiley, #3 is wearing glasses, #4 has a fat nose, #5 is wearing granny glasses - many more are possible using keys I don't have on my typewriter, such as "greater than" and "less than" signs for joking and frowning faces.) SMILEYS are placed at the end of a comment which, if said in person, would need to be softened by a smile. It can be difficult to determine when something printed is meant to be a joke or meant to be serious, whether someone is good-naturedly mouthing off or mounting a serious attack, and the presence (or absence!) of a SMILEY is considered a significant piece of information. Example: "That person should be sentenced to a year of real-time programming in COBOL! :-)"

Andrea Frankel
San Diego, California



SPECTRUM MICRODRIVES

I have been using Spectrum Microdrives on my TS2068 since May and I am very pleased with the results. To use the Microdrives on a TS2068 you need the following.

1) The ZX Expansion System package. This package contains Interface 1, a microdrive unit, a connector cable, and a wallet containing four microdrive cartridges. The cartridges consist of

- (a) A blank cartridge.
- (b) A cartridge with demonstration programs.
- (c) A business cartridge with Tasword 2 and Masterfile
- (d) A games cartridge with Ant Attack and a games designer utility.

I picked up my Microdrive expansion system in Britain last December for £99. However it is available in the U.S.A from the English Micro Connection, 15 Kilburn Court, Newport, Rhode Island 02840, (Tel: 401-849-3805) for \$129.95

2) The second item required is a Microdrive adapter. This reconfigures the TS2068 back connector to the Spectrum back connector configuration. Although advertised as a Microdrive adapter it can be used to connect other Spectrum hardware to your TS2068. The Microdrive adapter is available from The English Micro Connection for \$35.

3) Finally you need to make your TS2068 operate in the Spectrum mode. I'm told that you can use the Microdrives with a Spectrum ROM, a G Russell Romswitch, or one of Douglas Dewey's Emulators. However so far I can only confirm success with an Emulator. I've been unable to get the Microdrives working with a Spectrum ROM or a romswitch I made following the instructions in the December 1984 CATS newsletter. I do understand that other people have had more success using the Microdrives with an Emulator. Spectrum ROMs, Romswitches, and Emulators are available from several sources including The English Micro Connection.

To use the Microdrives the Microdrive adapter is plugged into the back of your TS2068, then Interface 1 is connected to the adapter and finally the

Microdrive itself is connected via it's cable to Interface 1.

The obvious advantages of the Microdrives are mass data storage and fast program find and loading. The longest loading time is about 30 seconds for a full 48K program. It is also a big advantage to be able to just type in a program name and let the computer find it in a matter of seconds rather than searching through a cassette tape.

Each Microdrive cartridge consists of an endless loop of tape about 1/8th inch wide which is capable of storing some 85K. to 90K. of data after it has been formatted.

I've been able to transfer some programs to microdrive from tape using a tape called 'Trans Express' from Romantic Robot. However I've only been able to transfer about 30% of my software this way. I have recently obtained another transfer program from LERM of England, but this tape requires a lot of tape manipulation to get programs onto Microdrives. I've been told that there is now a hardware fix available called Interface 3 (NOT made by Sinclair) which plugs into the back of Interface 1 and enables all programs to be transferred to Microdrive cartridges.

Problems have been few so far. One Microdrive cartridge has jammed. I solved this by pulling on the tape but I did lose one program on the cartridge. On another cartridge one file would not load after I had erased some files even though the CATALOGUE command said it was there.

I can thoroughly recommend the Microdrive system. It does work and it is available now, something which is not true of all of the other mass storage alternatives now being offered. It is cheaper to set up than disk based systems and competitive with other stringy floppy systems. If Interface 3 turns out to be as good at transferring programs to Microdrive as is claimed then this will give the Microdrives a big advantage

TONY BROOKS
972-4541

P.S. This text was prepared using a Microdrive based version of MSCRIPT driving a 2040 printer in 'fat characters' mode.

MODULO!

When we first learned long division, we discovered that not all division problems came out evenly. For that first year, we were told to report the undivided portion as the remainder. In the next year, a solution was offered, along with the introduction of fractions - just add a fraction created from the remainder and the divisor. With the introduction of decimals, we learned how to continue the division to any desired accuracy - a system that our computers follow even now. After the introduction of decimal division, it may have seemed as though the remainder was baby stuff, to be left behind forever.

T'aint true, however. In "adult" mathematics, the lowly remainder reappears with the concept of "modulus." It is a useful concept, with a number of applications in programming, and I'll discuss its use and implementation here.

How to Get It

On some computers, there is a function, MOD, that will return the remainder of a given division problem. It is invoked by LET N=x MOD y. This would set N equal to the remainder of x divided by y. While the Timex does not have a MOD function, we can determine when the MODULO equals zero by comparing (x/y) with INT (x/y). If x can be evenly divided by y, (x/y) will equal INT (x/y), as (x/y) will have no decimal component.

If the actual value of the remainder is needed, we can calculate this remainder: since the decimal fraction represents the remainder divided by the divisor, by multiplying the divisor back in again, we can "reconstitute" the remainder. First we isolate the decimal fraction, using ((x/y)-INT (x/y)); then we multiply by the divisor: "*y." The formula in BASIC that does this is: LET N=((x/y)-INT (x/y))*y.

Where to Use It

In fact, the Timex uses modulo in two places; RND is generated using a formula that involves modulo calculations, and TAB is reduced MOD 32 (try PRINT TAB 50; "HI"). Neither application can be adapted to other uses, however. To do that, we'll have to use the formula that appeared above.

Let's start at the bottom, with modulo 2. Try:

```
10 FOR X=1 TO 10
20 PRINT X,(X/2)-INT (X/2))*2
30 NEXT X
```

You should get:

```
1 1
2 0
3 1
4 0.... etc. Note that
```

even numbers Modulo 2 are equal to zero. We now have a way to sieve even from odd!

Now change the "2" in the above program to "3".

You should now get:

```
1 1
2 2
3 0
4 1
5 2
6 0.... etc.
```

By changing the number, you can construct a formula that will return to zero at any regular interval.

There are a number of useful intervals that can be used. By looking at a series modulo 12, you might keep track of months, but pause for a yearly summary every twelve months.

A major use of modulo is in converting between number bases. At times when dealing with the Timex, you will deal with Binary, Decimal, Hexadecimal, and what one writer has dubbed 256-imal. If you would like to convert numbers less than 256 to hexadecimal, you must reduce the number modulo 16. To find the two byte values for addresses in the Z-80 environment, the decimal address must be reduced modulo 256. To re-assemble the two bytes into a single decimal number, you must multiply the high-order byte by 256, and add it to the low-order byte. Since the low-order byte represents the remainder of a division, it is not multiplied by anything. An example is calculating RAMTOP on the 1000: PRINT PEEK 16388+256* PEEK 16389.

If you look over past programs, you will see a number of lines that contain the (x/y)-INT(x/y) idea. Each of these is using the modulo, that, as the remainder, you thought was left behind in fourth grade!

Thanks to The Algorithm Corner, in June '85 Computer Trader Magazine, for the idea.

MF

Continued from p. 6

If you know how big the machine code module is in bytes, use that number. If it is much smaller than 37080, it will reduce your loading time considerably.

Read over appendix E of the manual if you want additional information. I also want to thank Mark Fischer for his helpfulness and patience in answering my questions on this subject.

Ward Seguin

BINARY NUMBERS

$$128, 64, 32, 16, 8, 4, 2, 1/0$$

POSITIONAL VALUES OF 8 BITS IN 1 BYTE

$$64 + 16 + 2 + 1 = 83$$

$$01010011$$

READING BINARY

- 83 ÷ 64 = 1, remainder 19
- 19 ÷ 16 = 1, remainder 3
- 3 ÷ 2 = 1, remainder 1
- 1 ÷ 1 = 1, remainder 0
- = 83

WRITING BINARY

$$01010011 \quad 83$$

$$+ 01100101 \quad 101$$

$$= 10111000 \quad = 184$$

128 + 32 + 16 + 8

$$\begin{array}{r} 1 \quad 1 \quad 0 \\ + 1 \quad 0 \quad 0 \\ \hline (1) \quad 0 \quad 1 \quad 0 \\ \uparrow \text{CARRY} \end{array}$$

MAXIMUM VALUE OF ONE BYTE IS TOTAL OF ALL POSITIONAL VALUES OR 255

WRITING BINARY FROM DECIMAL
FIND POSITIONAL VALUE THAT IS HIGHEST BUT LESS THAN THE DECIMAL NUMBER. SUBTRACT AND FIND NEXT HIGHEST VALUE THATS LOWER THAN THE REMAINDER CONTINUE UNTIL TOTAL VALUE EQUAL DECIMAL NUMBER. WRITE EIGHT BIT BINARY NUMBER BY WRITING "1" AT POSITION OF EACH VALUE USED AND "0" AT POSITIONS SKIPPED.

V. CONGER

J. CONGER

SUBTRACTION (101 - 83 = 18)

$$01100101 = 101 \text{ decimal}$$

$$+ 10101101 = 83$$

$$(1) 00010010$$

16 + 2 = 18

IGNOR. (NOT A TRUE CARRY)

SUBTRACTION REQUIRES USE OF "2'S COMPLEMENT". EACH BIT IS REVERSED: 1=0 and 0=1. THEN 1 IS ADDED TO THE "COMPLEMENT" - AS ABOVE. THAT NUMBER IS THEN ADDED TO THE FIRST NUMBER AND ANY "CARRY" IS IGNORED. (SEE ZAKS ON PROBLEMS OF "SIGNS" AND "OVERFLOWS".)

SUBTRACTION WITH NEGATIVE ANSWER:

$$01010011 \quad 83$$

$$+ 10011011 \quad -101$$

$$01101110 = -18$$

NEGATIVE SIGN

$$00010001 = \text{COMPLEMENT}$$

$$+ 1$$

$$00010010$$

16 + 2 = 18 (-)

0110 0101 = 101 decimal
1001 1010 = "COMPLEMENT"
1001 1011 = 2's COMPLEMENT OF 101 decimal

WITH NEGATIVE ANSWER, IT MUST BE CONVERTED BY THE 2'S COMPLEMENT TO SHOW CORRECT ANSWER, AS AT LEFT.

(FOR MORE EXAMPLES, SEE ZAKS, P. 32)

Media Owner to Buy Sinclair Research

By Michael Schrage
Washington Post Staff Writer

Sinclair Research Ltd., a leading British personal computer company and the corporate brainchild of inventor Clive Sinclair, will be acquired by British media owner Robert Maxwell in a deal valued at about \$15.2 million.

The purchase brings together two of Britain's most unusual and best-known businessmen: Maxwell, the outspoken Czechoslovakian emigré and war hero who has built a multimillion-dollar communications empire in his adopted country, and Sinclair, the balding, bespectacled and entrepreneurial "boffin" who built the world's first under-\$100 computer and won the admiration of the conservative Thatcher government plus a knighthood.

The deal, completed this week-end in London, came with Sinclair Research just weeks away from insolvency, according to sources close to the company. As with other computer manufacturers around the world, Sinclair has been badly hurt by a drop in sales.

Last month, cash flow problems caused by excess computer inventory prompted the company to seek a two-month extension on \$12 million worth of debt payments to major suppliers Thorn EMI Ltd. and Timex. Sinclair recently was forced to start looking for a new managing director for his company.

According to a Maxwell spokesman, Sinclair will be a "life president" of the company "but he won't have an executive position." Maxwell is looking for a new managing director for the company.

Maxwell "believes in Sir Clive's inventive genius," said the spokesman, "but he's failed in the past because he doesn't understand commercial marketing things."

Maxwell is chairman of Pergamon Press Ltd., The British Printing & Communication Corp., Mirror Group Newspapers and Rediffusion Cablevision, Britain's largest cable television company.

SUNDAY TIMES OF LONDON

Timex hits at Sinclair

6/2/85

by Jane Bird and Peter Kingston

TROUBLED Sinclair Research has suffered a further blow with the discovery that Timex, the subcontractor which assembles most of its computers, is selling them abroad - and undercutting it on price.

Timex believes that its contract allows it to do this if Sinclair falls behind with payments. The Dundee-based assembler and Sinclair's other prime contractor, Thorn EMI, are together believed to be owed around £10m, though both have agreed to extend credit by two months.

Timex has approached Zeta Marketing of Manchester and asked it to dispose of an initial 65,000 Spectrums overseas. Zeta is offering these at a trade price of £71, around £16 less than Sinclair. The machine retails for £129.95.

Potential investors in Sinclair are expected to demand the introduction of some strong new senior management. The company is already looking for a chief executive so that Sir Clive could step down from the post and remain as chairman. But troubleshooting chief executives for troubled micro companies are proving hard to find - Acorn has been looking for several months without success. One obvious candidate, Robb Wilmot, has already joined forces with Sinclair for an ambitious chip project. The possibility that he might accept a full-time post has been denied by Wilmot, who is chairman of ICL, Britain's largest main-frame computer maker.

New Product Announcements

P. Hargrave

PO Box 1537

Ladysmith, BC

CANADA V0R 2E0

All prices include first class shipping.

CELESTIAL NAVIGATION: A program to reduce sextant sights and plot position anywhere in the world. Specifies best stars for a fix out of 192 in memory. Requires only angles and accurate time (and a 2068). By L. Benson. \$34.00 CAN \$25.00 US

STOCKS and LOANS: Two applications programs. STOCKS assists decisions in option trading. LOANS gives cost of borrowing info; total paid, monthly payments, etc. By F. Hamersveld. \$19.50 CAN \$14.50 US

COMPETITION: A business simulation program. Compete with the computer, or up to five other players. Includes graphs and financial statements to chart your progress. By P. Hargrave. \$19.50 CAN \$14.50 US

MILLIPEDE: An adaption of a certain arcade game for the 2068. Kid tested! By P. Hargrave. \$15.00 CAN \$11.25 US

Integrated Data Systems Peter Mc Mullin
30 Brookmont Rd. 2340 Queen St. E.
Toronto, Ont. Toronto, Ont.
Canada, M4L 3N1 Canada, M4E 169

CENTRONICS I/F for the 1000, and Word Sinc II.5: Updated 80 column version of Word Sinc II.4. Additional features include: custom fonts, multiple TABs and outdents; more. By Peter McMullin. \$30.00 CAN + \$1.00 s&h

Curry Computer
5344 W. Banff Ln.
Glendale, AZ 85306

A variety of programs for ZX/81, 2068, Spectrum, and QL. Includes Miner '49er, Lode Runner, Deus ex Machina, and GraphiQL. Write for catalog & prices.

Capitol Area Timex/Sinclair Users' Group
P.O.Box 725
Bladensburg, MD 20710

July 1985

Name _____

Address _____

_____ ZIP _____

Phone Home _____ Office _____

memberships - \$15.00 (family/individual); make checks payable to C.A.T.S.

If family membership, please list family members participating:

Occupation _____

Ham Radio call sign _____

Equipment

ZX 80 _____ RAM size _____

MA 80 _____ full keyboard _____

ZX 81 _____ Printer _____

ES 1000 _____ type _____

ES 2000 _____ other interface _____

Special interest use for computer: ie, games, ham radio interface,

business, other, etc. _____

Languages: Basic _____ Other _____

Machine _____

No. of years computer experience _____

What committees would you like to serve on? _____

Comments: Where did you hear about C.A.T.S.?

Do not write below:

Dt. Pd. _____ Amt. _____ Membership No. _____

Ca. _____ Ck. _____

DATED MATERIAL

Dues = \$15.00 per year, per family.

IF YOU ARE NOT A MEMBER OF CATS, THIS IS THE ONLY ISSUE YOU WILL RECEIVE

New Carrollton Public Library
7414 Riverdale Road, New Carrollton, MD

The next meeting of C.A.T.S. will be held on:
Saturday, July 13, 1985, at 2:00 PM to 5:00 PM

ANNANDALE
4375 GREENBERRY LANE
QUINTERO, MANUEL A.
301 VA 22003



Mar 8 2-5 PM
± 40 people

25164
4032
512
192
132

BULK RATE
U.S. POSTAGE
PAID
Permit No. 40
RANDALLSTOWN, MD.

CATS Newsletter
P.O. Box 725
Bladensburg MD 20710

The mailing address of the Capitol Area Timex/Sinclair User's Group is:

Capitol Area Timex/Sinclair User's group
P.O. Box 725
Bladensburg, MD 20710

CATS is a non-profit special interest organization dedicated to serving the interests of those who own, use, or are interested in learning more about the Timex/Sinclair family of personal computers.

The official contact person for CATS is JULES GESANG:
301*922-0767

Meetings are held on the second Saturday of each month at 2 P.M. in the large meeting room of the New Carrollton Branch Public Library.

Ham Radio Network Information

QZX Net...Wednesdays, 9p.m. local time; 14.345 MHz NV4F NCS
Eastern Regional Sinclair Net...Sundays, 1600 Z; 7.245 MHz
KQ2F NCS